

BAGGING CLASSIFICATION TREES
UNTUK PREDIKSI RISIKO PREEKLAMPSIA
(Studi Kasus : Ibu Hamil Kategori Penerima Jampersal
di RSUD Dr. Moewardi Surakarta)

Moch. Abdul Mukid¹, Triastuti Wuryandari², Desy Ratnaningrum³, Restu Sri Rahayu⁴
^{1,2}Staf Pengajar Jurusan Statistika FSM UNDIP
^{3,4}Mahasiswa Jurusan Statistika FSM UNDIP

Abstract

Preeclampsia is a spesific pregnancy disease in which hypertency and proteinuria occurs after 20 weeks of pregnancy. Classification Trees is a statistical method that can be used to identify potency of expectant women suffering from preeclampsia. This research aim to predict the risk of preeclampsia based on some individual variables. They are parity, work status, history of hypertension of preeclampsia, body mass index, education and income. To improve the stability and accuracy of the prediction were used the Bootstrap Aggregating Classification Trees method. By the method, classification accuracy reach to 86%.

Keywords : Pre-eclampsia, Bagging CART, Classification Accuracy

1. Pendahuluan

Angka Kematian Ibu (AKI) merupakan salah satu indikator yang digunakan untuk melihat derajat kesehatan perempuan. Salah satu penyebab utama kematian ibu hamil di Indonesia di samping perdarahan adalah preeklampsia. Preeklampsia adalah penyakit spesifik pada kehamilan yaitu terjadinya hipertensi dan proteinuria pada wanita hamil setelah umur kehamilan 20 minggu. Preeklampsia terjadi pada sekitar 2% - 8% dari kehamilan. Pada preeklampsia berat, tekanan darah naik setidaknya 160 mmHg (sistolik) dan 110 mmHg (diastolik), atau keduanya^[7]. Oleh sebab itu agar risiko preeklampsia dapat dideteksi sejak dini maka perlu dikembangkan sebuah model klasifikasi yang didasarkan atas informasi-informasi historis pasien preeklampsia.

Salah satu metode statistika yang dapat digunakan untuk mengklasifikasikan pasien ibu hamil kedalam resiko preeklampsia atau tidak adalah *Classification and Regression Trees* (CART). Metode ini merupakan metode yang dapat diterapkan pada data dengan ukuran obyek dan variabel yang besar^[1]. Jika variabel terikatnya bersifat kontinu maka pohon yang dihasilkan disebut dengan pohon regresi (*regression tree*), namun jika variabel terikatnya adalah kategorik maka pohon yang dihasilkan disebut dengan pohon klasifikasi (*classification tree*). Namun demikian, hasil pohon klasifikasi (*Classification Tree*) cenderung tidak stabil, karena perubahan-perubahan kecil pada data *learning* akan mempengaruhi hasil akurasi prediksi. Untuk memperbaiki stabilitas dan kekuatan prediksi pohon klasifikasi dapat digunakan metode *Bootstrap Aggregating (Bagging) Classification Tree*^[2].

2. Tinjauan Pustaka

2.1. Pengertian Preeklampsia

Menurut American College of Obstetricians and Gynecologist (ACOG) preeklampsia adalah hipertensi (tekanan darah sistolik ≥ 140 mmHg dan diastolik ≥ 90 mmHg) disertai proteinuria (≥ 30 mg/liter urin atau ≥ 300 mg/24 jam) yang didapatkan

setelah umur kehamilan 20 minggu. Preeklampsia berat melibatkan kelainan sistem multiorgan. Pasien yang terkena mungkin memiliki keluhan seperti sakit perut, disfungsi sistem saraf pusat (sakit kepala, ensefalopati, penglihatan kabur, atau kebutaan), edema paru, hipoksia atau sebagai manifestasi lain dari kebocoran kapiler. Preeklampsia merupakan komplikasi dari kehamilan yang berhubungan dengan tingginya morbiditas dan mortalitas maternal serta pertumbuhan janin terhambat^{[7],[4]}.

2.2. Classification and Regression Trees (CART)

CART adalah sebuah metode yang termasuk dalam anggota analisis klasifikasi pohon keputusan, karena proses analisis dari CART digambarkan dalam bentuk atau struktur yang menyerupai sebuah pohon. Jika variabel terikatnya bersifat kontinu maka pohon yang dihasilkan disebut dengan pohon regresi (*regression tree*), namun jika variabel terikatnya adalah kategorik maka pohon yang dihasilkan disebut dengan pohon klasifikasi (*classification tree*). Untuk selanjutnya akan dijelaskan hanya pada pembentukan pohon klasifikasi saja.

Teknis kerja dari CART dalam membuat sebuah pohon klasifikasi dikenal dengan istilah *binary recursive partitioning*. Prosesnya disebut *binary* karena setiap simpul induk akan selalu mengalami pemilahan kedalam tepat dua simpul anak. Sedangkan *recursive* berarti bahwa proses pemilahan tersebut akan diulang kembali pada setiap simpul anak sebagai hasil pemilahan sebelumnya, sehingga simpul anak tersebut sekarang menjadi simpul induk. Proses pemilahan ini akan terus dilakukan sampai tidak ada kesempatan lagi untuk melakukan pemilahan berikutnya. Istilah *partitioning* berarti bahwa *learning sample* yang dimiliki dipilah ke dalam bagian-bagian yang lebih kecil^[5].

Kriteria pemilahan didasarkan pada nilai-nilai dari variabel bebas yang dimiliki. Misalkan variabel terikat y yang bertipe kategorik dianggap berkaitan dengan M variabel bebas x_1, x_2, \dots, x_M . Proses *binary recursive partitioning* dapat diilustrasikan sebagai proses pembagian atau penyekatan dari ruang berdimensi M dari variabel-variabel x kedalam subruang-subruang yang saling bebas dan tidak tumpang tindih. Pertama, pilih satu variabel bebas x_m dan sebuah nilai dari x_m misalkan s_1 untuk memilah ruang berdimensi M tadi kedalam dua subruang. Subruang bagian pertama berisi obyek-obyek dengan $x_m \leq s_1$ sedang subruang kedua berisi obyek-obyek dengan nilai $x_m > s_1$. Kemudian masing-masing dari subruang tadi dipilah kembali dengan cara yang sama oleh sebuah variabel bebas dengan nilai tertentu. Variabel bebas yang terpilih dapat x_m kembali atau variabel bebas lainnya. Proses ini terus berlanjut sampai diperoleh subruang-subruang yang lebih sedikit anggotanya. Proses pemilahan akan terus berlanjut sampai diperoleh pohon klasifikasi yang maksimal.

Terdapat berbagai kriteria yang dapat digunakan untuk menyeleksi pemilah terbaik di masing-masing simpul^[1]. Dua diantaranya yang sering digunakan adalah pemilah yang mengurangi indeks ketidakmurnian (*impurity index*) pada simpul ke- t , yaitu $i(t) = -\sum_{j=1}^k p(j|t) \log p(j|t)$ atau pemilah yang mengurangi indeks keberagaman Gini (*Gini index of diversity*) pada simpul ke- t , yaitu $i(t) = \sum_{i \neq j} p(i|t)p(j|t)$. Jika sebuah pemilah s dari simpul t membagi simpul t ke dalam simpul t_R dengan proporsi p_R dan simpul t_L dengan proporsi p_L maka pengurangan nilai keragaman/ketidakmurnian didefinisikan dengan

$$\Delta i(s, t) = i(t) - p_R i(t_R) - p_L i(t_L)$$

Suatu pemilah s^* akan digunakan untuk memilah simpul t menjadi dua buah simpul t_R dan t_L jika s^* memaksimalkan nilai $\Delta i(s, t)$ atau

$$\Delta i(s^*, t) = \max_s \Delta i(s, t)$$

Proses pemilahan atau pembentukan pohon klasifikasi akan berhenti apabila hanya terdapat satu obyek di dalam simpul terakhir atau adanya batasan minimum n . Semua obyek yang berada di dalam sebuah simpul merupakan anggota yang memiliki karakteristik yang cenderung homogen. Pohon klasifikasi yang terbentuk sebagai hasil dari proses ini dinamakan *maximal tree* atau *largest tree* (T_{\max})^[5].

Pohon klasifikasi yang dibentuk melalui proses pemilahan secara rekursif akan berukuran sangat besar. Hal ini disebabkan karena aturan penghentian yang digunakan hanya berdasarkan pada banyaknya obyek pada simpul terminal atau besarnya penurunan tingkat keragaman dalam tiap simpul anak hasil pemilahan. Semakin banyak pemilahan yang dilakukan maka tingkat kesalahan prediksi juga akan semakin kecil. Namun pohon klasifikasi yang terbesar atau maksimal terlalu sulit untuk dipahami sehingga menyebabkan *overfitting* untuk data baru. Masalah tersebut diatasi dengan melakukan pemangkasan pada pohon klasifikasi maksimal untuk mendapatkan pohon klasifikasi dengan ukuran yang optimal^[5].

Langkah awal pemangkasan dilakukan terhadap T_1 , yaitu suatu subpohon dari pohon maksimal T_{\max} . Untuk mendapatkan T_1 dari T_{\max} , diambil t_L dan t_R yang merupakan simpul anak kiri dan simpul anak kanan dari T_{\max} yang dihasilkan dari pemilahan pada setiap simpul induk t . Jika $R(t) = R(t_L) + R(t_R)$ dengan $R(t) = \left(1 - \max_j p(j|t)\right)p(t)$ maka simpul anak t_L dan t_R tersebut dipangkas. Proses ini diulang terus sampai suatu pohon klasifikasi optimal diperoleh.

2.3. Proses Pelabelan Kelas (*Class Assignment*)

Pelabelan kelas dilakukan mulai dari awal pemilahan simpul hingga simpul akhir terbentuk, karena setiap simpul yang dibentuk memiliki kesempatan menjadi simpul akhir. Pelabelan tiap simpul akhir berdasarkan aturan jumlah anggota kelas terbanyak yaitu jika:

$$P(j_0|t) = \max_j P(j|t) = \max_j \frac{N_j(t)}{N(t)}$$

maka label kelas untuk simpul akhir t adalah j_0 . Dimana $p(j_0|t)$ adalah proporsi kelas j_0 pada simpul t , $p(j|t)$ adalah proporsi kelas j pada simpul t , $N_j(t)$ adalah banyak pengamatan kelas j pada simpul t dan $N(t)$ adalah jumlah pengamatan pada simpul t .

2.4. *Bootstrap Aggregating (Bagging)*

Bootstrap Aggregating merupakan salah satu teknik yang diusulkan oleh Breiman yang dapat digunakan pada beberapa metode klasifikasi dan regresi untuk mereduksi variansi dari suatu prediktor, dan dengan demikian dapat memperbaiki kualitas prediksi^[8]. *Bootstrap* merupakan suatu *resampling* atau pengambilan data sampel yang saling bebas dan dilakukan secara berulang-ulang, yang digunakan untuk menduga tingkat kesalahan (*error*) dari pengulangan tersebut^[3]. Sampel *bootstrap* diperoleh dengan cara mengambil sampel secara acak dengan pemulihan dari sampel asli.

Proses pembuatan dugaan secara *bagging* menggunakan pohon adalah sebagai berikut^[6]:

1. Tahapan *bootstrap*

- a. Ambil sampel acak dengan pemulihan berukuran n dari gugus data *learning*.
- b. Susun pohon terbaik berdasarkan data tersebut.
- c. Ulangi langkah a s/d b sebanyak B kali sehingga diperoleh B buah pohon klasifikasi.

2. Tahapan *aggregating*

Lakukan prediksi gabungan berdasarkan B buah pohon klasifikasi tersebut dengan menggunakan aturan *majority vote* (suara terbanyak).

Penggunaan *bagging* ini sangat membantu terutama mengatasi sifat ketidakstabilan pohon klasifikasi dan pohon regresi. Pada banyak gugus data yang dicoba, *bagging* mampu mengurangi tingkat kesalahan klasifikasi pada kasus klasifikasi^[2]. Mengenai berapa banyak pengulangan *bootstrap* yang diperlukan, studi Breiman menunjukkan bahwa menggunakan 50 kali untuk kasus klasifikasi dan 25 kali untuk kasus regresi dapat memberikan hasil yang memuaskan^[2].

3.5. Ketepatan Klasifikasi

Apparent Error Rate (APER) merupakan suatu ukuran yang digunakan untuk menyatakan nilai proporsi sampel yang salah diklasifikasi^[4]. Jika n_1 merupakan jumlah anggota observasi yang diklasifikasikan $Y = 0$ dan n_2 merupakan jumlah anggota observasi yang diklasifikasikan $Y = 1$, maka bentuk matriks konfusinya yaitu seperti pada Tabel 1.

Tabel 1. Matriks Konfusi		
Observasi	Prediksi	
	Kelas 0	Kelas 1
Kelas 0	n_{00}	n_{01}
Kelas 1	n_{10}	n_{11}

$$APER = \frac{n_{01} + n_{10}}{n_0 + n_1}$$

dengan n_{00} = jumlah amatan kelas 0 yang tepat diklasifikasikan sebagai kelas 0

n_{01} = jumlah amatan kelas 0 yang salah diklasifikasikan sebagai kelas 1

n_{11} = jumlah amatan kelas 1 yang tepat diklasifikasikan sebagai kelas 1

n_{10} = jumlah amatan kelas 1 yang salah diklasifikasikan sebagai kelas 0

Maka, untuk menghitung nilai ketepatan klasifikasi adalah $1 - APER$.

3. Metode Penelitian

3.1. Data dan Variabel Penelitian

Data yang digunakan dalam penelitian ini merupakan data sekunder, yang terdiri dari 250 pasien preeklampsia dan 250 pasien tidak preeklampsia dengan responden adalah ibu hamil usia 20-35 tahun di RSUD Dr. Moewardi Surakarta tahun 2013. Variabel respon (Y) yang dianalisis adalah status pasien yang dikategorikan sebagai berikut:

$Y = 0$, untuk ibu hamil yang menderita preeklampsia

$Y = 1$, untuk ibu hamil yang tidak menderita preeklampsia.

Peubah penjelas dalam penelitian ini terdiri dari delapan variabel (X) yaitu paritas, pekerjaan, riwayat hipertensi, riwayat preeklampsia, Indeks Massa Tubuh (IMT), pendidikan, dan pendapatan.

3.2. Langkah-langkah Analisis Data

Alat analisis utama yang digunakan dalam penelitian ini adalah *Bootstrap Aggregating Classification Trees* dengan tahap-tahap sebagai berikut:

1. Analisis deskriptif data nasabah berdasarkan status pasien.
2. Menentukan proporsi data *learning* dan data *testing* yaitu dengan perbandingan 80% : 20%.
3. Dari 80% data *learning*, kemudian lakukan *resampling* secara acak dan dengan pemulihan.
4. Pembentukan pohon klasifikasi dengan tahap-tahap sebagai berikut :
 - a. Proses pemilahan simpul (*splitting nodes*) dengan kriteria pemilahan terbaik (*goodness of split*) untuk memilah/memecah sebuah simpul.
 - b. Proses pelabelan kelas (*class assignment*) dengan aturan jumlah anggota terbanyak pada simpul t .
 - c. Proses penghentian pembentukan pohon klasifikasi (*stop the splitting*).
 - d. Proses pemangkasan pohon klasifikasi (*Pruning*), yaitu jika dua simpul anak dan simpul induk memenuhi persamaan $R(t) = R(t_L) + R(t_R)$.
5. Prediksi terhadap data *testing*.
6. Proses *bagging* (*bootstrap aggregating*).

Langkah 1: Ulangi langkah 3 s/d 5 dengan data *learning* yang berbeda sehingga diperoleh 25 buah pohon klasifikasi.

Langkah 2: Melakukan pendugaan gabungan berdasarkan hasil pendugaan 25 buah pohon klasifikasi yang terbentuk menggunakan aturan *majority vote* (suara terbanyak).
7. Menghitung ketepatan klasifikasi hasil dari pendugaan gabungan menggunakan kriteria APER.

4. Hasil Dan Pembahasan

4.1. Deskripsi Data

Persentase ibu hamil dengan status pasien preeklampsia dan tidak preeklampsia yang terlibat dalam penelitian ini masing-masing sebesar 250, sehingga ukuran sampel keseluruhannya ada 500 responden. Tabel 2 berikut ini adalah tabulasi silang antara variabel-variabel bebas dan variabel terikat yaitu status preeklampsia responden.

4.2. Pembagian Data

Dari keseluruhan amatan yaitu sebanyak 500, diambil sebanyak 400 amatan (80%) sebagai data *learning* yang digunakan untuk proses pembentukan pohon klasifikasi dan sebanyak 100 amatan (20%) sebagai data *testing* yang digunakan untuk mengukur kemampuan dari pohon klasifikasi yang terbentuk dalam memprediksi status pasien.

Dalam *Bootstrap Aggregating* terdapat dua proses. Pertama, proses *bootstrap* untuk pembentukan pohon klasifikasi dilakukan sebanyak 25 kali pengulangan dengan banyak data *learning* pada masing-masing pohon klasifikasi 400, yang diambil secara acak dan dengan pemulihan. Kedua, proses *aggregating* yaitu prediksi gabungan berdasarkan data *testing* pada masing-masing pohon klasifikasi yang telah terbentuk dengan aturan *majority vote* (suara terbanyak).

Tabel 2. Tabulasi Silang antara Variabel-Variabel Bebas dan Variabel terikat

Variabel Bebas	Kategori	Variabel Terikat	
		Preeklampsia	Tidak Preeklampsia
Status Paritas	Multigravida	165	136
	Primigravida	85	114
Status Bekerja	Bekerja	114	165
	Tidak Bekerja	136	85
Riwayat Hipertensi	Hipertensi	65	0
	Tidak Hipertensi	185	250
Riwayat Preeklampsia	Ya	35	0
	Tidak	215	250
Indeks Masa Tubuh	Kegemukan	17	34
	Normal	233	216
Tingkat Pendidikan	SD	85	51
	SMP	102	85
	SMA	63	114
Penghasilan	≤ 500.000	170	145
	> 500.000	80	105

4.3. Pembentukan Pohon Klasifikasi Pertama

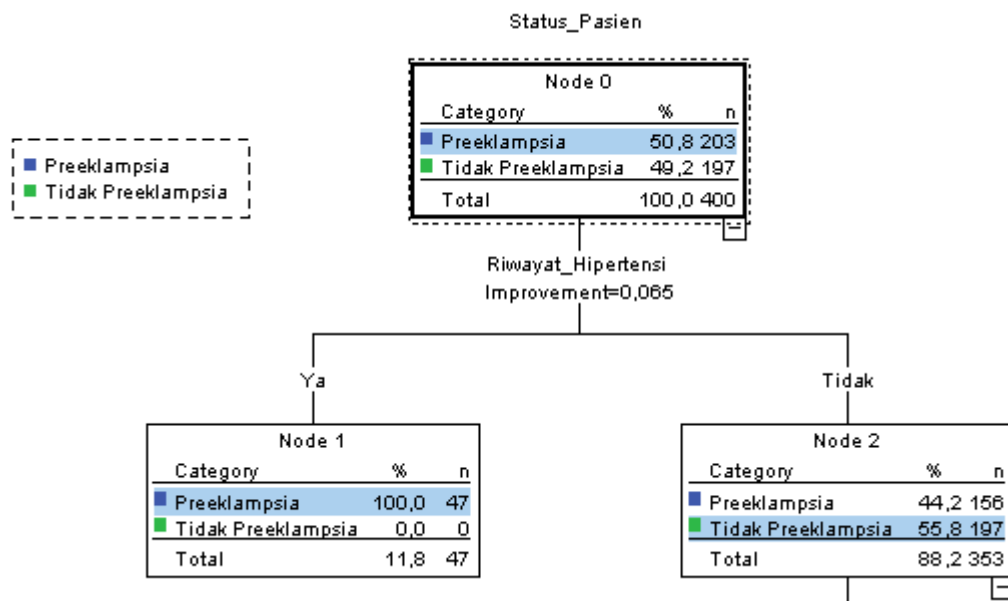
4.3.1. Proses Pemilahan Simpul (*Splitting Node*)

Suatu *split* s akan digunakan untuk memilah simpul t menjadi dua buah simpul yaitu simpul kiri (t_L) dan simpul kanan (t_R) jika s memaksimalkan nilai $\Delta i(s^*, t) = \max_{s \in S} \Delta i(s, t)$. Nilai *goodness of split* pada semua kemungkinan pemilah dapat dilihat pada Tabel 3.

Tabel 3. *Goodness of Split*

No.	Variabel	Pemilah		<i>Goodness of Split</i>
		Simpul Kiri	Simpul Kanan	
1.	Paritas	Paritas 1 s/d 4	Primigravida	0,008
2.	Status Bekerja	Tidak Bekerja	Bekerja	0,024
3.	Riwayat Hipertensi	Ya	Tidak	0,065
4.	Riwayat Preeklampsia	Ya	Tidak	0,028
5.	IMT	<i>Overweight</i>	Normal	0,016
		SD	SMP, SMA	0,008
6.	Pendidikan	SMP	SD, SMA	0,004
		SMA	SD, SMP	0,024
7.	Pendapatan	≤ 500.000	> 500.000	0,009

Pemilah terbaik untuk simpul 0 atau simpul akar adalah variabel riwayat hipertensi dengan kriteria pemilahan Ya pada simpul kiri (simpul 1) dan Tidak pada simpul kanan (simpul 2). Variabel tersebut terpilih karena memiliki nilai *goodness of split/improvement* tertinggi dari variabel lainnya. Proses pemilahan dapat dilihat pada Gambar 1.



Gambar 1. Pemilahan Simpul Akar pada Pohon Klasifikasi Pertama

4.3.2. Proses Pelabelan Kelas (*Class Assignment*)

Proses pelabelan kelas pada simpul-simpul yang terbentuk berdasarkan aturan jumlah anggota kelas terbanyak yaitu jika $P(j_0|t) = \max_j P(j|t)$, maka $j_0 = j$ dengan $j = 1$ (preeklampsia) dan $j = 0$ (tidak preeklampsia). Sebagai contoh yaitu simpul 2 pada Gambar 1.

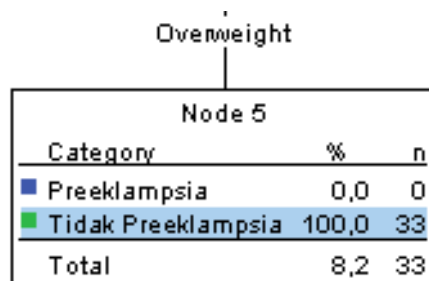
$$P(\text{Preeklampsia}|\text{simpul 2}) = \frac{156}{353} = 0,442$$

$$P(\text{Tidak Preeklampsia}|\text{simpul 2}) = \frac{197}{353} = 0,558$$

sehingga simpul 2 diberi label kelas tidak preeklampsia, karena proporsi kelas tidak preeklampsia lebih besar dari proporsi kelas preeklampsia.

4.3.3. Proses Penghentian Pemilahan (*Stop the Splitting*)

Pohon klasifikasi maksimal yang pertama memiliki 6 simpul dalam dan 7 simpul akhir. Proses penghentian pemilahan dapat dilihat pada contoh simpul 5 Gambar 2. Pada simpul 5 terdapat 33 amatan pada kelas yang sama (homogen) sehingga proses pemilahan simpul dihentikan.



Gambar 2. Simpul 5 pada Pohon Klasifikasi Maksimal Pertama

4.3.4. Proses Pemangkasan Pohon Klasifikasi (*Pruning*)

Proses pemangkasan pohon klasifikasi maksimal dimulai dengan mengambil t_L yang merupakan simpul kiri dan t_R yang merupakan simpul kanan dari T_{max} yang dihasilkan dari simpul induk t . Jika diperoleh dua simpul anak dan simpul induk yang memenuhi persamaan $R(t) = R(t_L) + R(t_R)$, maka simpul anak t_L dan t_R dipangkas. Proses tersebut diulang sampai tidak ada lagi pemangkasan yang mungkin dilakukan. Sebagai contoh simpul yang dipangkas yaitu pada simpul 10. Pada simpul 10 diperoleh

$$r(\text{simpul } 10) = 1 - \max_j P(j|\text{simpul } 10) = 1 - 0,769 = 0,231$$

$$P(\text{simpul } 10) = \frac{121}{400} = 0,302$$

$$R(\text{simpul } 10) = r(\text{simpul } 10) * P(\text{simpul } 10) \\ = 0,231 * 0,302 = 0,07$$

Selanjutnya dihitung nilai $R(t_L)$ dan $R(t_R)$ pada simpul anak, yaitu simpul 11 dan simpul 12. Pada simpul 11 diperoleh

$$r(\text{simpul } 11) = 1 - \max_j P(j|\text{simpul } 11) = 1 - 0,714 = 0,286$$

$$P(\text{simpul } 11) = \frac{98}{400} = 0,245$$

$$R(\text{simpul } 11) = r(\text{simpul } 11) * P(\text{simpul } 11) \\ = 0,286 * 0,245 = 0,07$$

Pada simpul 12 diperoleh

$$r(\text{simpul } 12) = 1 - \max_j P(j|\text{simpul } 12) = 1 - 1 = 0$$

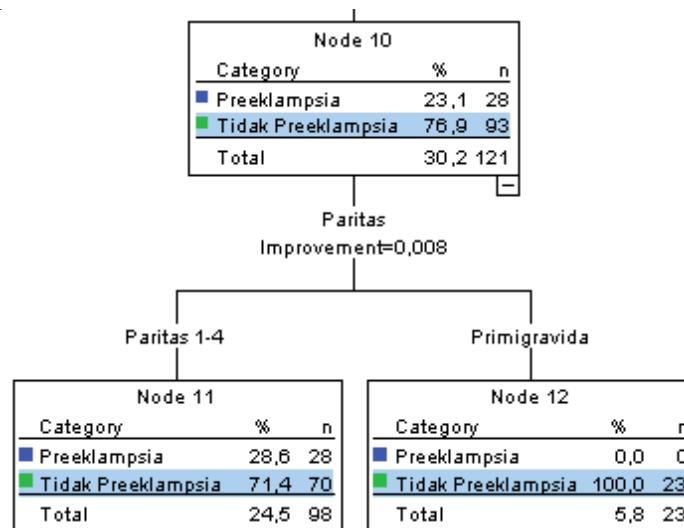
$$P(\text{simpul } 12) = \frac{23}{400} = 0,058$$

$$R(\text{simpul } 12) = r(\text{simpul } 12) * P(\text{simpul } 12) \\ = 0 * 0,058 = 0$$

$$\text{Dengan demikian } R(\text{simpul } 11) + R(\text{simpul } 12) = 0,07 + 0 \\ = 0,07$$

$$= R(\text{simpul } 10)$$

Karena $R(\text{simpul } 11) + R(\text{simpul } 12) = R(\text{simpul } 10)$ maka simpul 11 dan 12 dipangkas.



Gambar 3. Simpul 10 pada Pohon Klasifikasi Maksimal Pertama yang Dipangkas

4.4. Prediksi pada Pohon Klasifikasi Pertama

Sebagai contoh prediksi pada data *testing* pertama dengan kriteria paritas = paritas 1 s/d 4, status bekerja = tidak bekerja, riwayat hipertensi = ya, riwayat preeklampsia = ya, IMT = *overweight*, pendidikan = SD, dan pendapatan ≤ 500.000 . Kriteria tersebut masuk pada model Riwayat_Hipertensi = Ya yang diprediksi masuk ke dalam kelas preeklampsia.

4.5. Prediksi dengan Majority Vote

Setelah dilakukan proses *bootstrap* sebanyak 25 kali dilanjutkan dengan pembentukan pohon klasifikasi dan prediksi data *testing* pada masing-masing pohon klasifikasi, langkah selanjutnya adalah prediksi gabungan (*aggregating*) berdasarkan 25 prediksi pada data *testing* dengan aturan *majority vote* (suara terbanyak). Sebagai contoh pada data *testing* yang pertama seperti pada bagian 4.4.

Berdasarkan hasil prediksi dari 25 pohon klasifikasi, data *testing* pertama yang diprediksi masuk kelas preeklampsia sebanyak 25 sedangkan yang diprediksi masuk kelas tidak preeklampsia sebanyak 0. Hasil akhir prediksi berdasarkan suara terbanyak, sehingga prediksi akhir data *testing* pertama masuk ke dalam kelas preeklampsia.

4.6. Ketepatan Klasifikasi

Hasil prediksi gabungan yang telah diperoleh selanjutnya digunakan untuk menguji ketepatan klasifikasi pada penerapan metode *bagging Classification Trees*. Uji ketepatan klasifikasi dilakukan menggunakan matriks konfusi pada Tabel 4.

Tabel 4. Matriks Konfusi Hasil Klasifikasi Menggunakan *Bagging CART*

Observasi	Prediksi	
	Preeklampsia	Tidak Preeklampsia
Preeklampsia	32	14
Tidak Preeklampsia	0	54

$$APER = \frac{n_{01} + n_{10}}{n_0 + n_1} = \frac{14 + 0}{100} = 0,14$$

Dengan nilai APER = 0,14, maka ketepatan klasifikasinya adalah $1 - 0,14 = 0,86$ atau sebesar 86%.

5. Kesimpulan

Berdasarkan hasil dari pembahasan, dapat disimpulkan bahwa metode *bagging Classification Trees* dapat diterapkan untuk mengklasifikasikan pasien ibu hamil dengan risiko preeklampsia atau tidak. Dalam penelitian ini, pengklasifikasian pasien ibu hamil dengan metode *bagging Classification Trees* menghasilkan ketepatan klasifikasi sebesar 86%.

DAFTAR PUSTAKA

1. Breiman, L., J. H. Friedman, R. A. Olshen, and C. J., Stone. *Classification and Regression Trees*, Chapman and Hall (Wadsworth, Inc.), New York, 1993.
2. Breiman, L., Bagging Predictors, *Machine Learning*, 1996, Vol. 24: 123-140
3. Efron, B. and Tibshirani, R. J., *An Introduction to the Bootstrap*, Chapman & Hall, New York, 1993
4. Elosha E, Chike N, *et al.*, Preeclampsia 2012, *Journal of Pregnancy*, Meharry Medical College, Nashville, USA, 2012: 1-4
5. Lewis, R. J., *An Introduction to Classification and Regression Trees (CART) Analysis*, Presented at the 2000 Annual Meeting of Society for Academic Emergency Medicine of Sanfransisco. California, 2000.
6. Sutton, C. D., Classification and Regression Trees, Bagging, and Boosting. *Handbook of Statistics*, 2005, Vol. 24: 303-329.
7. Winfred, W. Williams, Jr., *et al.*, 2005. *Case 38-2005: A 29-Year Old Pregnant Woman with the Nephrotic Syndrome and Hypertension*. New England J. Of Medicine, case records of the Massachusetts general hospital: 2593